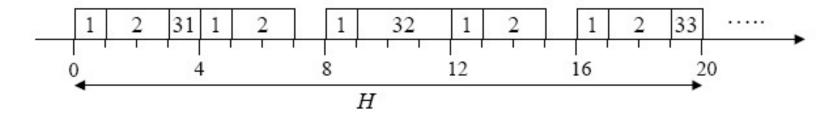
Manca ancora qualcosa: Come allocare i job ai frame?



- L'allocazione manuale dei job ai frame è plausibile solo per pochi task periodici
- In generale si tratta di trovare una schedule che <u>assegna</u> i job ai frame compatibili con i vincoli temporali
- Il problema è trattabile e ammette infinite soluzioni se la preemption è ovunque consentita
- Esempio:

$$T1 = (4,1,4)$$
 $T2 = (5,2,7)$ $T31 = (20,1,20)$ $T32 = (20,3,20)$ $T33 = (20,1,20)$

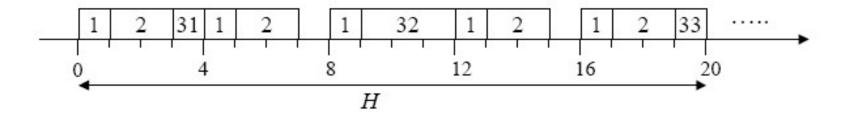


Manca ancora qualcosa: Come allocare i job ai frame?



- L'allocazione manuale dei job ai frame è plausibile solo per pochi task periodici
- In generale si tratta di trovare una schedule che <u>assegna</u> i job ai frame compatibili con i vincoli temporali
- Il problema è trattabile e ammette infinite soluzioni se la preemption è ovunque consentita
 Che succede se invede la preemption non è consentita? Che tipo di problema diventa?

T1 = (4,1,4) T2 = (5,2,7) T31 = (20,1,20) T32 = (20,3,20) T33 = (20,1,20)



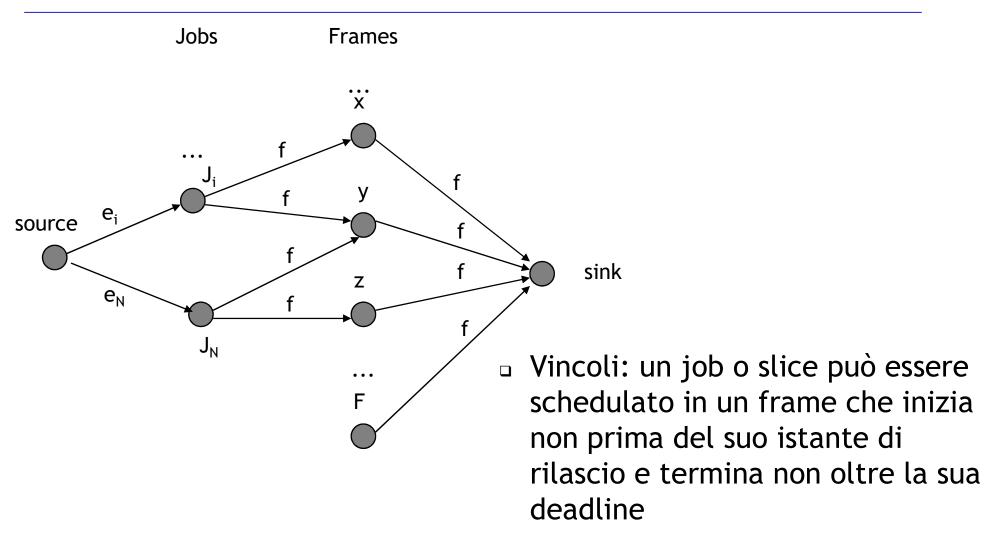
Algoritmo per sintesi di schedule statiche



- Nel caso generale il problema è NP-hard
- Se la preemption è consentita ovunque diventa polinomiale e può essere risolto con un algoritmo che calcoli il flusso massimo su un grafo
- Struttura del grafo:
 - un vertice per ogni job o slice J_i(d_i,e_i) all'interno del ciclo maggiore: {J_i, i=1,...,N}
 - □ un vertice per ogni frame: $\{f_j, j=1,...,F\}$
 - vertici source e sink per il calcolo del flusso max
 - \Box lati (J_i, f_j) di capacità f se J_i può essere schedulato in f_j
 - lati da source a J_i di capacità e_i
 - \Box lati da f_j a sink di capacità f

Struttura del grafo di flusso





Algoritmo per sintesi di schedule statiche /1



Inizializzazione:

- Calcola tutte le possibili dimensioni di frame che soddisfano i vincoli:
 - $\Box \ [H/f] H/f = 0$
 - \Box 2f MCD(f,p_i) \leq D_i \forall i
- □ Il vincolo $f \ge e_i \forall i$ può essere ignorato (per possibile partizionamento dei job)
- Iterazione:
- A partire dal valore massimo di f, per la dimensione di frame corrente costruisci il grafo di flusso ed esegui un algoritmo per il calcolo del flusso massimo

Algoritmo per sintesi di schedule statiche /2



- □ Il flusso massimo possibile, in base alla struttura del grafo, è soggetto al vincolo $\Phi \leq \sum_{i=1,N} e_i$
- La schedule esiste (e viene costruita dall'algoritmo) se il flusso massimo calcolato è uguale alla somma dei tempi di esecuzione: $\Phi_{\text{max}} = \sum_{i=1,N} e_i$
- Se un job è eseguito su più frame, significa che esso deve essere partizionato in sottojob

Algoritmo per sintesi di schedule statiche



 Flusso massimo del grafo: formulazione e risoluzione come problema LP o risoluzione tramite algoritmo dei cammini aumentati

Algoritmo dei cammini aumentati



- 0. Trasforma il grafo orientato in non orientato; la capacità è annotata in prossimità del vertice di partenza del lato.
- 1. Trova un cammino che aumenta il flusso totale, cioè un cammino diretto tra source e sink tale che ogni lato del cammino abbia una capacità > 0.
- 1.b Se il cammino non esiste, il flusso identificato è già ottimo.
- 2. Identifica il lato del cammino con capacità residua minima c*. Aumenta il flusso del cammino di c*.
- 3. Diminuisci di c* la capacità residua di ogni lato lungo il cammino. Aumenta di c* la capacità residua di ogni lato del cammino in direzione opposta. Ritorna al passo 1.

Algoritmo dei cammini aumentati



- Al termine dell'algoritmo:
 - se $\Phi_{\text{max}} = \sum_{i=1,N} e_i$ l'algoritmo ha determinato una schedule fattibile
 - l'insieme dei flussi dei lati dai vertici che rappresentano i job ai vertici che rappresentano i frame costituisce una schedule fattibile preemptive

Sintesi di schedule statiche



- \Box Esempio: [T1=(4,1), T2=(5,2,7), T3=(20,5)]
- Troviamo (ad es.) la schedule: H=20, m=4, con f1=[J11(1),J21(2)]; f2=[J12(1),J311(3)]; f3=[J13(1),J22(2)]; f4=[J14(1),J23(2),J312(1)]; f5=[J15(1),J24(2),J313(1)]
- Altri esempi per esercizi:

[T1=
$$(4,3)$$
, T2= $(6,1.5)$]
[T1= $(4,1)$, T2= $(5,2,5)$, T3= $(20,5)$]

Esempio Liu, p. 86:

$$[T1=(4,1), T2=(5,1.8), T3=(20,1), T4=(20,2)]$$

Foglio di lavoro



Foglio di lavoro/2



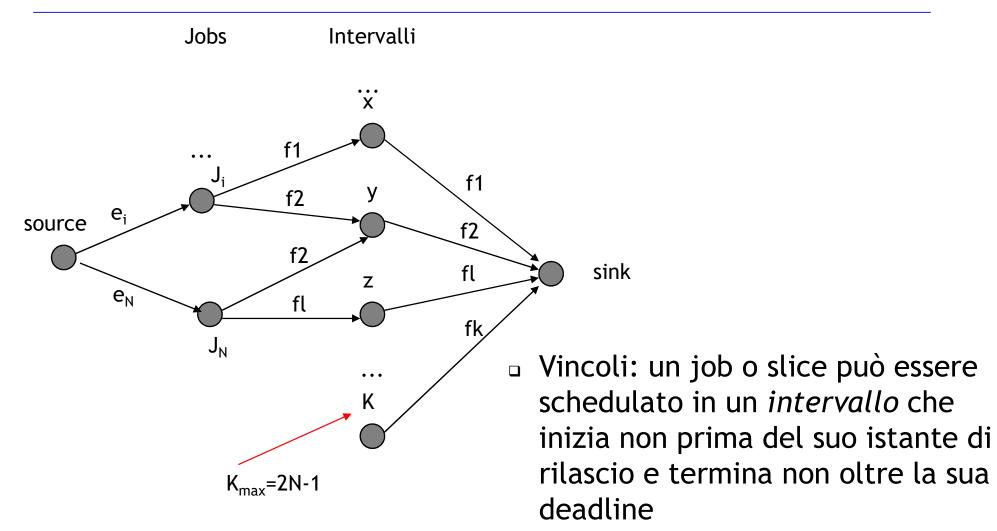
Sintesi di schedule per task non periodici



- La formulazione tramite calcolo del flusso massimo del problema di scheduling può essere generalizzata per ricavare schedule di job aperiodici indipendenti con istanti di rilascio e deadline arbitrari
- Gli istanti di rilascio e le deadline di N job dividono il tempo in al più 2N-1 intervalli
- Nel grafo è sufficiente sostituire i vertici dei frame con vertici
 I_i che rappresentano ciascuno degli intervalli
- La capacità entrante ed uscente di ciascun vertice I_j è pari alla dimensione dell'intervallo
- □ Se $\Phi = \sum_{i=1,N} e_i$ la schedule *preemptive* rappresentata dai flussi è fattibile

Grafo di flusso per schedulazione aperiodica





Esercizio



- Per il seguente insieme di job aperiodici, determinare una schedule mediante algoritmo dei cammini aumentati
- Ji=(di,ci,ri): J1=(12,3,0), J2=(8,3,0), J3=(14,4,6), J4=(6,2,2)

Assegnamento



- Per i job dell'esercizio precedente ricavare anche le schedule prodotte dagli algoritmi EDF e LRT
- Ricercare le schedule prodotte dagli algoritmi FIFO, SJF, LJF (con preemption), verificandone la fattibilità
- Inviare soluzione in unico file pdf entro Lunedì 20/3 ore 12.30,
 con oggetto email:
 - SORT2023 Esercizi Lezione 17/3